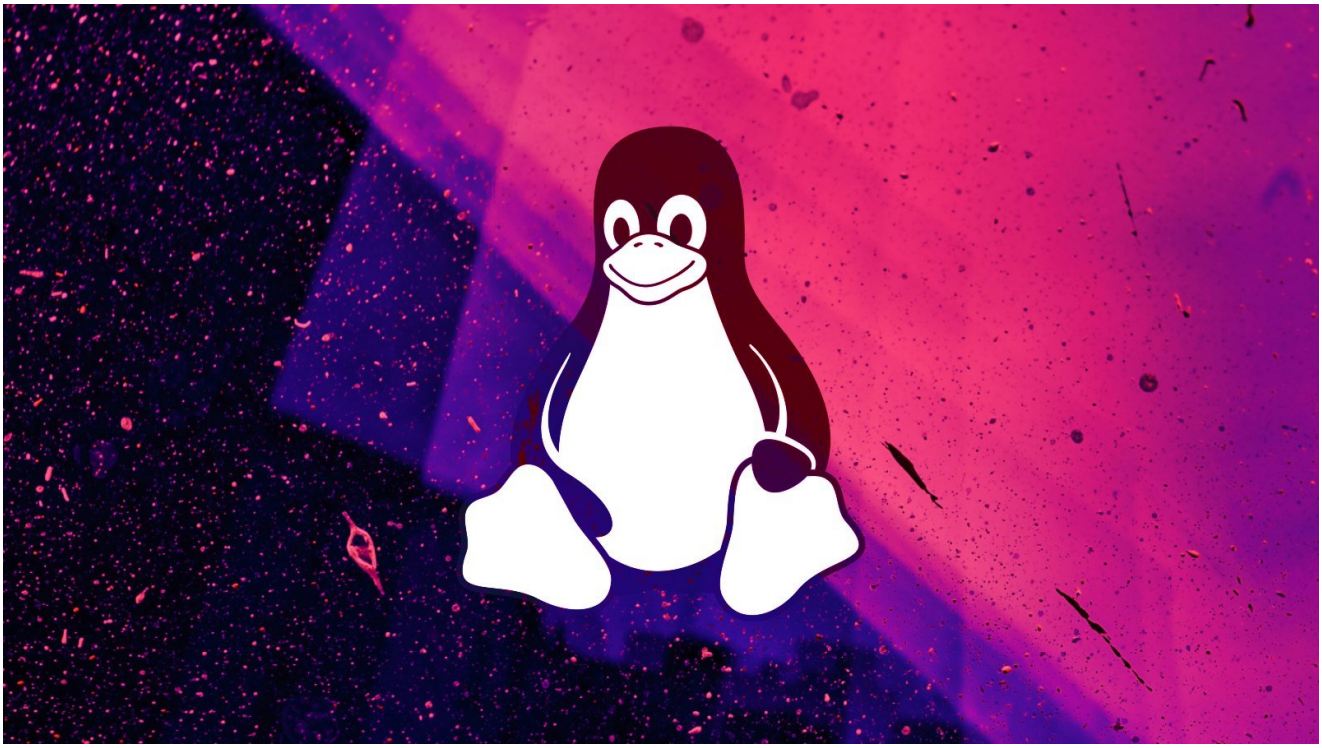
Stealthier version of Linux BPFDoor malware spotted in the wild

bleepingcomputer.com/news/security/stealthier-version-of-linux-bpfdoor-malware-spotted-in-the-wild/

Bill Toulas

By
Bill Toulas

- May 11, 2023
- 06:02 PM
- 0



A new, stealthier variant of the Linux malware 'BPFDoor' has been discovered, featuring more robust encryption and reverse shell communications.

BPFDoor is a stealthy backdoor malware that has been active since at least 2017 but was only discovered by security researchers around 12 months ago.

The malware gets its name from the use of the 'Berkley Packet Filter' (BPF) for receiving instructions while bypassing incoming traffic firewall restrictions.

BPFDoor is designed to allow threat actors to maintain lengthy persistence on breached Linux systems and remain undetected for extended periods.

New BPFDoor version

Until 2022, the malware used RC4 encryption, bind shell and iptables for communication, while commands and filenames were hardcoded.

The newer variant analyzed by Deep Instinct features static library encryption, reverse shell communication, and all commands are sent by the C2 server.

	"New stealthy" 2023 variant	"Old" 2022 variant
Encryption	Static library encryption	RC4 Encryption
Communication	Reverse-Shell	Bind shell and iptables
Commands	No hardcoded commands – all commands are sent through the reverse-shell	Hardcoded commands
Filenames	Not hardcoded	Hardcoded

Differences between the old and new versions *(Deep Instinct)*

By incorporating the encryption within a static library, the malware developers achieve better stealth and obfuscation, as the reliance on external libraries like one featuring the RC4 cipher algorithm is removed.

The main advantage of the reverse shell against the bind shell is that the former establishes a connection from the infected host to the threat actor's command and control servers, allowing communication to the attackers' servers even when a firewall protects the network.

Finally, removing hardcoded commands makes it less likely for anti-virus software to detect the malware using static analysis like signature-based detection. It theoretically also gives it more flexibility, supporting a more diverse command set.

Deep Instinct reports that the latest version of BPFDoor is not flagged as malicious by any available AV engines on VirusTotal, despite its first submission on the platform dating February 2023.

Operation logic

Upon first execution, BPFDoor creates and locks a runtime file at "/var/run/initd.lock," and then forks itself to run as a child process, and finally sets itself to ignore various OS signals that could interrupt it.

Signal Number	Signal Name	Signal Description
1	SIGHUP	SIGHUP ("signal hang-up") is a signal sent to a process when its controlling terminal session is closed.
2	SIGINT	SIGINT ("signal interrupt") is a signal sent when a user interrupts a program (Ctrl + C)
3	SIGQUIT	SIGQUIT is a signal sent to terminate a process.
13	SIGPIPE	SIGPIPE is a signal sent when a pipe breaks.
17	SIGCHLD	SIGCHLD is a signal sent when a child process exits.
21	SIGTTIN	SIGTTIN is a signal sent to a process attempting to read from the same terminal session and is blocked.
23	SIGTTOU	SIGTTOU is a signal sent to a process attempting to write to the same terminal session and is blocked.

OS signals the malware is set to ignore *(Deep Instinct)*

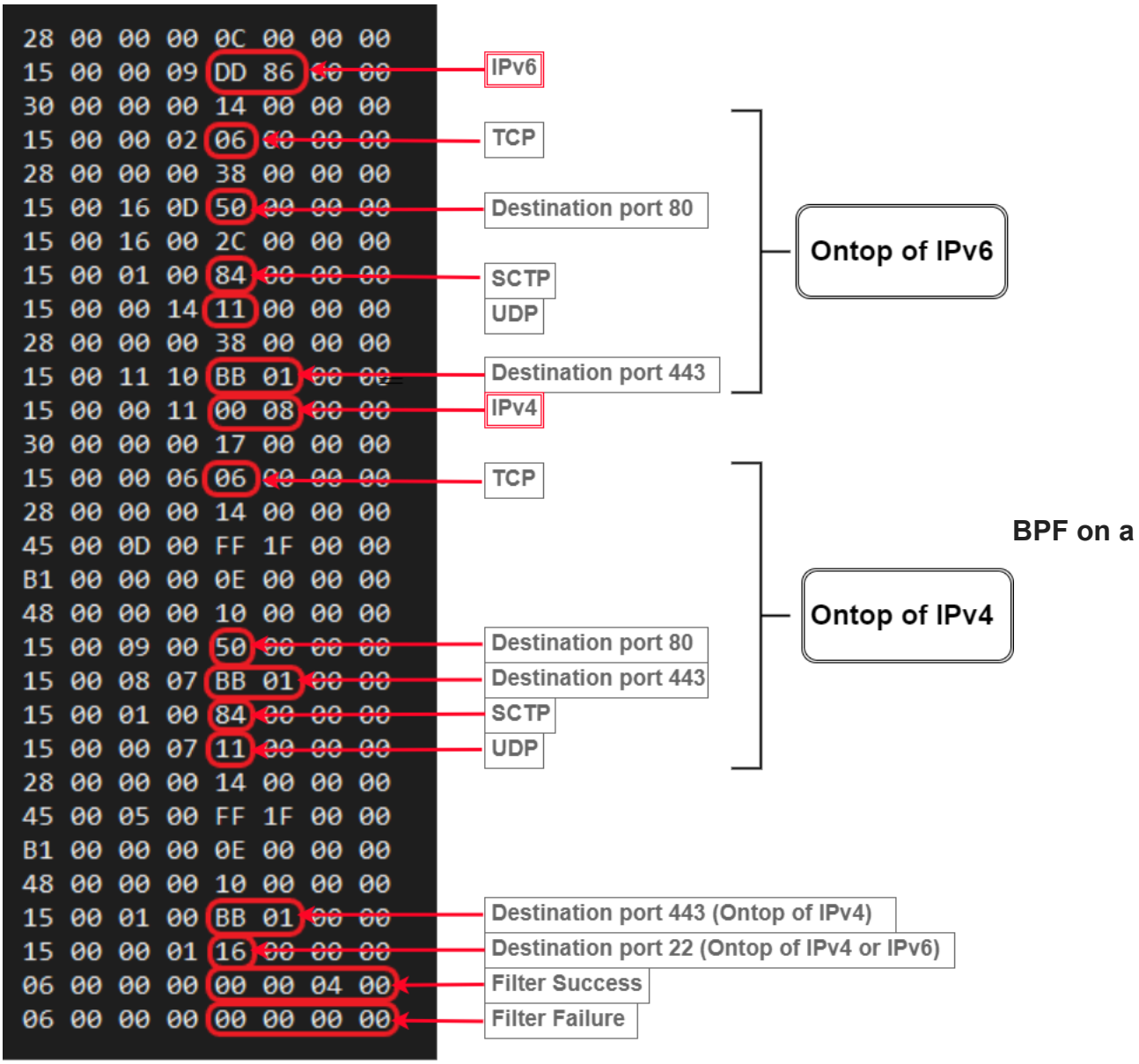
Next, the malware allocates a memory buffer and creates a packet sniffing socket that it'll use for monitoring incoming traffic for a "magic" byte sequence ("\x44\x30\xCD\x9F\x5E\x14\x27\x66").

```
while ( 1 )
{
  while ( 1 )
  {
    while ( (DWORD)recvfrom(fd_sock, recved_buff, 0x10000, 0, 0, 0) < 0 )
      ;
    v8 = recved_buff[14];
    ++packet_counter;
    v9 = 4 * (v8 & 0xF);
    if ( v9 > 0x13 )
    {
      v10 = (DWORD)&recved_buff[v9 + 14];
      if ( 4 * ((unsigned byte)recved_buff[v9 + 26] >> 4) > 0x13 )
      {
        v11 = _byteswap_ulong(*(DWORD *)&recved_buff[v9 + 22]);
        if ( _byteswap_ulong(*(DWORD *)(v10 + 4)) == 0x4430CD9F && v11 == 0x5E142766 )
          break;
      }
    }
  }
}
```

Looking for the magic byte sequence *(Deep Instinct)*

At this stage, BPFDoor attaches a Berkley Packet Filter to the socket to read only UDP, TCP, and SCTP traffic through ports 22 (ssh), 80 (HTTP), and 443 (HTTPS).

Any firewall restrictions present on the breached machine won't impact this sniffing activity because BPFDoor operates at such a low level that they're not applicable.

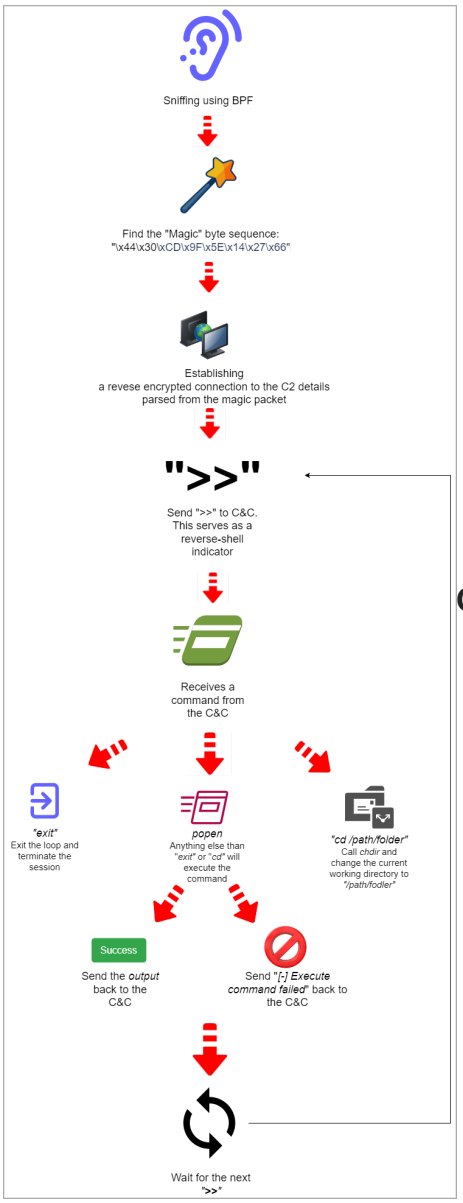


socket *(Deep Instinct)*

"When BPFdoor finds a packet containing its "magic" bytes in the filtered traffic, it will treat it as a message from its operator and will parse out two fields and will again fork itself," explains Deep Instinct.

"The parent process will continue and monitor the filtered traffic coming through the socket while the child will treat the previously parsed fields as a Command & Control IP-Port combination and will attempt to contact it."

After establishing a connection with the C2, the malware sets up a reverse shell and waits for a command from the server.



Operational diagram

(Deep Instinct)

BPFDoor remains undetected by security software, so system admins may only rely on vigorous network traffic and logs monitoring, using state-of-the-art endpoint protection products, and monitor the file integrity on "/var/run/initd.lock."

Also, a May 2022 report by CrowdStrike highlighted that BPFDoor used a 2019 vulnerability to achieve persistence on targeted systems, so applying the available security updates is always a crucial strategy against all types of malware.

Related Articles:

New PowerExchange malware backdoors Microsoft Exchange servers

RomCom malware spread via Google Ads for ChatGPT, GIMP, more

'Operation Magalenha' targets credentials of 30 Portuguese banks

Cybercrime gang pre-infects millions of Android devices with malware

VirusTotal AI code analysis expands Windows, Linux script support

Bill Toulas

Bill Toulas is a technology writer and infosec news reporter with over a decade of experience working on various online publications. An open source advocate and Linux enthusiast, is currently finding pleasure in following hacks, malware campaigns, and data breach incidents, as well as by exploring the intricate ways through which tech is swiftly transforming our lives.